

Feature Selection and Ensemble of Regression models for Predicting the Protein Macromolecule Dissolution Profile

Varun Kumar Ojha*, Konrad Jackowski*, Ajith Abraham*, Václav Snášel*

*IT4Innovations, VŠB Technical University of Ostrava, Ostrava, Czech Republic

varun.kumar.ojha@vsb.cz, konrad.jackowski@pwr.edu.pl, ajith.abraham@ieee.org, vaclav.snasel@vsb.cz

Abstract—Predicting the dissolution rate of proteins plays a significant role in pharmaceutical/medical applications. The rate of dissolution of Poly Lactic-co-Glycolic Acid (PLGA) micro- and nanoparticles is influenced by several factors. Considering all factors leads to a dataset with three hundred features, making the prediction difficult and inaccurate. Our present study consists of three phases. Firstly, dimensionality reduction techniques are applied in order to simplify the task and eliminate irrelevant and redundant attributes. Subsequently, a heterogeneous pool of several classical regression algorithms is created and evaluated. Regression algorithms in the pool are independently trained to identify the problem at hand. Finally, we test several ensemble methods in order to elevate the accuracy of the prediction. The Evolutionary Weighted Ensemble method proposed in this paper offered the lowest RMSE and significantly outperformed competing classical algorithms and other ensemble techniques.

Keywords-feature selection; regression models; ensemble; protein dissolution;

I. INTRODUCTION

Predicting the dissolution profile of Poly (Lactic-co-Glycolic Acid) (PLGA) micro and nanoparticles is a complex problem as there are several potential factors that influencing the dissolution of PLGA protein particles [1]. A collection of the influencing factors counted upto three hundred. Therefore, the primary approach one may adopt is to reduce the dimensionality of the dataset. Dimensionality reduction techniques transform a high dimension dataset into a low dimensional datasets, thereby improving model's computational speed, predictability and generalization ability.

In the present scope of the study, our focus on the PLGA nano- or microspheres dissolution properties and drug release rate. Szlkeket *et al.* [2] and Fredenberg *et al.* [3] described that the drug release from the PLGA matrix is mainly governed by two mechanisms, diffusion and degradation/erosion. Several factors influencing the diffusion and degradation rate of PLGA described by Kang *et al.* [4], [5], Blanco and Alonso [6] and Mainardes *et al.* [7] are pore diameters, matrix Active Pharmaceutical Ingredient (API) interactions, API - API interactions and formulation composition. Szlkeket *et al.* [2] offered a predictive model to describe the underlying relationship between those influencing factors on the drug release profile, where they focus on the feature selection, artificial neural network and genetic

programming to obtain a suitable prediction model for the said purpose. In the past, several mathematical models, including Monte Carlo and cellular automata microscopic models, were proposed by Zygourakis and Markenscoff [8] and Gpferich [9]. A partial differential equations model was proposed by Siepmann *et al.* [10] to address the influence of underlying PLGA properties on the drug release rate/protein dissolution.

We shall discuss the PLGA drug release problem and dataset collection mechanisms in section II-A. In section II-B, we shall discuss the computational tools available for reducing dimensionality. A concise discussion on the prediction models are given in section II-C, followed by a discussion on ensemble methods used for making a ensemble of regression models in section II-D. A comprehensive discussion on the experimental setup is offered in section III. Finally, we shall conclude our discussion in section IV.

II. METHODOLOGY

A. Problem Description

PLGA micro- and nanoparticles could play significant role in the medical application and toxicity evaluation of PLGA-based multiparticulate dosage form. PLGA microparticles are important diluents used to produce drugs in dosage form. Apart from playing a role as a filler, PLGA as an excipient, and alongside pharmaceutical APIs plays crucial role in various ways. It helps in the dissolution of drugs, thus increasing absorbability and solubility of drugs. It helps in pharmaceutical manufacturing process by improving APIs powder's flowability and nonstickiness. In addition, it helps in vitro stability, such as in the prevention of denaturation over expected shelf life.

A comprehensive experiment is conducted on the the dataset offered by Szlkeket *et al.* [2] in their article "Heuristic modeling of macromolecule release from PLGA microspheres". The dataset collected by Szlkeket *et al.* [2] from various literature has three hundred input features categorized into four groups, namely protein descriptor, plasticizer, formulation characteristics, and emulsifier. The formulation characteristics group contains features such as PLGA inherent viscosity, PLGA molecular weight, lactide-to-glycolide ratio, inner and outer phase Polyvinyl alcohol (PVA) concentration, PVA molecular weight, inner phase volume,

encapsulation rate, mean particle size, PLGA concentration and experimental condition (dissolution pH, number of dissolution additives, dissolution additive concentration and production method and dissolution time). The protein descriptor, plasticizer and emulsifier feature groups contains 85, 98 and 101 features respectively. The regression model sought to predict the dissolution percentage or solubility of PLGA, which depends on the features mentioned above. In order to avoid overfitting, collected data were preprocessed by adding noise to them. The dataset was then normalized in the range [-1.0, 1.0].

B. Feature Selection Tools

Feature selection techniques enable us to choose from the set of input features we have in our hand. Especially, feature selection become significant step towards development of a predication model where it requires expensive (both in time and cost) experimental examination.

1) *Backward Feature Elimination (BFE)*: The Backward Feature Elimination filter provided in the KNIME¹ is used for feature elimination. The basic methodology and principle behind BFE filter is to start with the maximum number feature in hand (in this case it starts with three hundred features) and eliminate features one by one in an iterative manner. At each iteration, the resulting accuracy of prediction is evaluated for all combinations of the remaining attributes and subset of attributes with the highest accuracy being propagated to the next iteration.

2) *Correlation Based Feature Selection (CFS)*: CFS assesses the value of group of attributes that concern the individual predictive ability of each features, as well with the possibility of repetition among the features [11].

3) *Classifier Based Feature Selection (Class)*: The classifier based feature selection method evaluates attribute subsets on training data and uses a classifier to estimate the merits of a set of attributes.

4) *Wrapper Feature Selection (Wrapper)*: The wrapper based feature selection evaluates attribute sets by using a learning scheme and cross-validation (cv) to estimate the accuracy of the learning scheme for a set of attributes [12].

C. Elementary Regression Models

The Regression/Prediction model tries to figure out the relationship between independent variables X (input) and dependent variables y (output). Moreover, it tries to find unknown parameters β such that error (2) is minimized given that

$$\hat{y} = f(X, \beta), \quad (1)$$

where \hat{y} is predicted output. Let $e_i = (\hat{y}_i - y_i)$ be the difference between the dependent variable y_i and predicted

value \hat{y}_i . Therefore, Root Mean Square Error (RMSE) ξ over data samples of size n is given as

$$\xi = \sqrt{\frac{1}{n} \sum_{i=1}^n e_i^2}. \quad (2)$$

Regression models such as Linear Regression (LReg), Gaussian Process Regression (GPReg), Multilayer Perceptron (MLP) and Sequential Minimal Optimization Regression (SMOReg) are used.

1) *Linear Regression (LReg)*: Linear regression is the simplest predictive model where independent variables ($|X| = n \times p$), dependent variable ($|y| = p$ with noise/error $|\varepsilon| = p$ may be written as

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^T \cdot \beta + \varepsilon_i. \quad (3)$$

2) *Gaussian Process Regression (GPreg)*: The Gaussian process describe by Rasmussen [13], [14] is fully specified by its mean function $m(x)$ and covariance function $k(x, x')$. This is a natural generalization of the Gaussian distribution whose mean m and covariance k is a vector and matrix respectively. The Gaussian distribution is over vectors, whereas the Gaussian process is over functions f . Therefore we may write

$$f \sim \mathcal{GP}(m, k). \quad (4)$$

3) *Multilayer Perceptron (MLP)*: Multilayer perceptron (MLP) is a feedforward neural network having one or more hidden layers in between the input and output layers [15], [16]. A neuron in a MLP first computes a linear weighted combination of real valued inputs and then limits its amplitude using a non-linear activation function.

In the present case, MLP is trained using the Backpropagation algorithm propounded by Rumelhart et al. [17] and the Resilient propagator (RProp) developed by Riedmiller et al. [18].

4) *REP Tree (REP)*: Reduced Error Pruning Tree (REP) is a fast decision tree learner. It builds a decision tree based on information gain or reducing the variance and prunes it using reduced-error pruning (REP) with overfitting [19], [20].

5) *Sequential Minimal Optimization Regression (SMOReg)*: Sequential minimal optimization (SMO), an algorithm for the training of Support Vector Regression (SVR) proposed by Smola and Schlkopf [21], [22], [23], was an extension of the SMO algorithm proposed by Platt [24] for the SVM classifier. SVR attempts to minimize the generalization error bound in order to achieve generalized performance. The idea of SVR is based on the computation of a linear regression function in a high dimensional feature space where the input data are mapped via a non-linear function.

¹KNIME - Professional Open-Source Software of KNIME.com AG

D. Ensemble regression algorithm

Searching for the best regression algorithm is not a trivial task. Apart from having plethora of options, listed in the previous section, one has to also decide, what are the optimal set of parameters for each of tested algorithms is. Usually, there are very little guidance on how to select an algorithm and adjust its parameters for the problem in hand. In such cases, experimental tests can help to make decision. Nonetheless, in many cases the obtained results are not satisfactory or even not acceptable. In such situations, the ensemble approach can be used. Basically, it relies on the assumption that the properly modelled fusion of responses of several elementary predictors shall produce more accurate results and reduce regression error. Formally, let Π be a set of k predictors given as

$$\Pi = \{f_1, f_2, \dots, f_k\}, \quad (5)$$

where, f_k indicate the state of the k^{th} predictor. Each of the predictors are trained independently. The ensemble system perform fusion of the outputs produced by the predictors in the set Π . In the simplest form, it can take form of simple average called Mean Output Regression (MOR), given as

$$\hat{F}(x) = \frac{1}{K} \sum_{k=1}^K f_k(x), \quad (6)$$

where \hat{F} is an ensemble system.

A natural advantage of this model is its simplicity as the output of the ensemble can easily be obtained by simple mathematical transformation without the necessity of setting any additional parameters. On the other hand, the main drawback of this model is that it treats all the elementary predictors equally regardless of their quality. Weak predictors affect the final signal in the same degree as the strong ones. As a result, the quality of the ensemble is close to the average of all its constituents. Better results can be obtained when the contribution of a particular predictor depends on its quality. Higher the accuracy of the predictor, higher its weight in ensemble. The ensemble method is therefore called Quality Weighted Output Regression (QWOR), given as

$$\hat{F}(x) = \frac{1}{K} \sum_{k=1}^K w_k f_k(x), \quad (7)$$

where $\sum_{k=1}^K w_k = 1$. In the simplest implementation, the weights shall be counter proportional to the RMSE of the given predictor. Although, in the more advanced algorithms, the weights can be set in the course of learning procedure, e.g., be application of evolutionary algorithms.

E. Diversity of the ensemble

There are several issues that have to be dealt with in order to make an application of ensemble approach effective.

One of the most essential is maintaining diversity among the predictors in the ensemble. Collecting a set of similar regression algorithms does not allow to take any advantage from their fusion. Diversity can be ensured by the application of one of the following procedures.

- 1) Collecting predictors based on different models.
- 2) Differentiate elementary predictor inputs.

In the first approach, it is assumed that different regression algorithms naturally make errors that are uncorrelated, even when they are trained on the same data. The second group consists of algorithms that create an ensemble based on the same regression model, but diversity is caused by training each of them on data partitions (as it exists in the range of Bagging algorithms) or using heterogeneous feature sets (techniques used in Random Subspace algorithms).

1) *Random Subspace (RS)*: Random Subspace is a method of constructing an ensemble of predictors where a pseudorandom procedure is used to select components of a feature vector separately for each ensemble constituent. The output of the ensemble is then obtained by averaging the outputs [25].

2) *Bagging*: Breiman [26] introduced the bagging methods which is basically a combination of multiple predictor. At first subsets are prepared by cutting the original dataset called bootstrapping. A sequence predictors allowed to run over the subsets of the dataset. Finally, the results from each of the predictors are aggregated using voting in order to get the final result. This method is supposed to enhance performance and reduce variances in order to improve [27], [28].

3) *Evolutionary weighted ensemble (EWE)*: EWE makes decision based on formula (6). The learning process search for set of weights values which minimizing RMSE of the ensemble. The learning set is used for that purpose the

Algorithm: Evolutionary weighted ensemble

```

Input: Learning set
S - population size
G - number of generations
Π - a set of individual predictors
repeat
  Initialize population
  for t = 1 to G do
    Evaluate Population over Learning Set
    Select Elite
    Select parents
    Mutation
    Crossover
    Create offspring population
  end for
until the stopping criterion satisfied.

```

Figure 1. Evolutionary weighted ensemble (EWE)

objective function for learning procedure can be written as

$$\text{RMSE}^{\hat{F}}(w_1, \dots, w_k) = \sqrt{\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K (w_k f_k(x_n, i) - y_n)^2}, \quad (8)$$

where x_n , and y_n denotes the n^{th} input-target pair in the learning set that consists of N samples.

We decided to apply the evolutionary based algorithm [29]. It processes population of possible solutions encoded in form of chromosomes which represent set of weights. An overview of the EWE training procedure is presented in Figure 1.

III. EXPERIMENTAL SETUP AND RESULTS

The Experimental setup for the feature selection and identification of the corresponding regression model for the prediction of protein molecules is as follows: The experiment is conducted using KNIME, MATLAB² and WEKA³. As mentioned in section II, the dataset obtained for the PLGA dissolution profile has three hundred features, therefore the primary objective is to reduce the dimension of the dataset. The feature selection techniques discussed in section II-B are used for the said purpose. Subsequently, elementary prediction models are employed and their performances is assessed using 10 cross-validation (10cv) sets. The selection of the prediction model is based on the average of the Root Mean Square Error (RMSE) computed over a set of 10 results. In the final part of our experiment, we explored ensemble methods in order to exploit the elementary regression/prediction models.

A. Feature selection methods results

After cleaning and preprocessing the dataset, it goes under a feature selection treatment in which we use the backward feature elimination technique with the GPreG prediction model with and RBF kernel, LReg, a three-layer MLP with fifty neurons at the hidden layer, a 0.3 learning rate, a 0.2 momentum rate, SMOReg with a polynomial kernel, a 0.001 epsilon value, a 0.001 tolerance label and REP. The combination of attributes which offer the lowest RMSE is termed as the optimal feature set. For example, the optimal feature set obtained using the regression models GPreG, LReg, MLP, SMOReg and REP are 18, 32, 31, 30 and 31 with the RMSE (result of normalized dataset) 0.143, 0.156, 0.121, 0.153 and 0.126 respectively. Hence, we may examine the performance of prediction models using the feature set with smaller attributes. In the present study, we have chosen features sets with 10, 5 and 1 attributes.

Now, when it comes to stochastic feature selection techniques, we have Correlation, Classifier and Wrapper based

²MATLAB is trademark of the MathWorks, Inc.

³WEKA - Data Mining Software in Java developed by the machine learning group at the University of Waikato, Free Software Foundation, Inc.

Table I
EXPERIMENTAL RESULTS FOR 10CV DATASETS PREPARED WITH DISTINCT RANDOM PARTITIONS OF THE COMPLETE DATASET USING FEATURE SELECTION TECHNIQUE (IDENTIFICATION OF REGRESSION MODEL) *Note. Values are average of 10 RMSE.*

Selection Method	Selected Features	GPreG	LReg	MLP	REP	SMOReg
No Selection	300	16.81	17.07	18.57	13.05	17.95
BFE	1	27.47	26.61	28.33	24.37	26.97
BFE	5	17.11	23.45	23.11	14.23	23.38
CFS	5	20.80	25.08	22.41	18.31	25.42
ClassMLPgreedy	7	17.96	25.03	22.26	14.96	25.35
BFE	10	15.93	19.98	21.00	13.19	19.53
ClassMLPbfs	15	15.88	22.90	16.83	13.91	24.23
WrapperGPgreedy	15	14.88	20.22	15.20	13.34	20.86
ClassGPbfs	16	18.46	23.07	19.71	14.19	23.69
ClassGPgreedy	19	15.06	19.05	15.61	14.03	19.68
WrapperMLPgreedy	19	16.44	24.01	20.42	14.26	24.85
WrapperLRgreedy	24	15.91	17.46	17.03	13.54	18.02
BFE	Optimal*	15.71	17.85	17.82	13.90	17.88
ClassLRbfs	31	15.95	16.92	15.63	14.00	17.58
ClassLRgreedy	37	16.31	17.14	16.27	14.02	17.69

*Optimal set of attributes for regression models GPreG, LReg, MLP, REP and SMOReg are 18, 32, 31, 31 and 30 respectively.

feature selection methods. These feature selection methods are used to determination the merits (predictability) of different combination of features. After assigning merits to several features sets obtained using aforementioned feature selection techniques, best first search (bfs) and greedy search (greedy) are used for selecting a desire optimum feature set. Interestingly, in the present problem, when we use CFS, both the bfs and greedy search produce identical feature sets containing 5 attributes. The Classifier based feature selection is patched with GPreG, MLP, and LR respectively in order to evaluate the merits of the feature set. Subsequently, bfs and greedy search are used to determine the optimum feature set. Therefore, we have ClassGPbfs⁴, ClassGPgreedy, ClassMLPbfs, ClassMLPgreedy, ClassLRbfs and ClassLRgreedy feature selection methods indicating a classifier based method with GPreG as a feature set merit evaluator and the bfs as method for the optimum feature set selection and so an. Similarly, the WrapperGPgreedy, WrapperMLPgreedy, WrapperLRgreedy indicates a combination of wrapper-based feature selection, where GPreG, MLP and LReg are used to evaluate feature set. Interestingly, both bfs and greedy search offers identical feature set. A list of feature selection methods and corresponding selected features appears in Table I.

B. Regression model results

In order to identify a suitable regression model, we have chosen several regression models available in WEKA. Selected regression models such as GPreG, LReg, MLP, REP

⁴ClassGPbfs i.e. Class+GPreG+bfs indicates that the classifier GPreG is used to determine the merits of the feature sets and best first search (bfs) is used to search the best feature set among them. Please follow a similar convention for the rest of the other abbreviations.

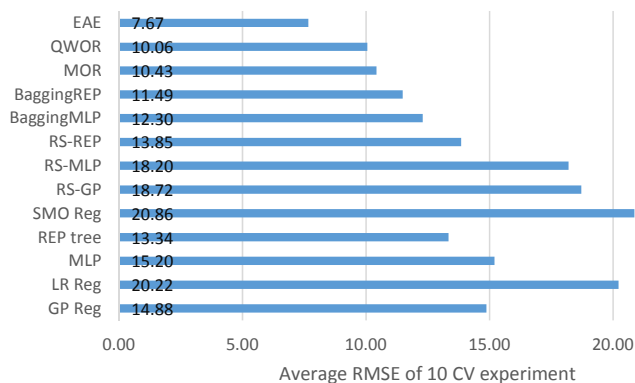


Figure 2. A comprehensive conclusion of the results obtained from each regression models including the ensemble techniques used. Note: We have selected a feature set obtained using wrapper+GPreg+Greedy search

and SMOReg are discussed in section II-C. The parameter setting corresponding to the mentioned regression models are as follows: GPReg has a radial basis function kernel, MLP has 10 neurons at the hidden layer and a 0.3 and 0.1 learning rate and momentum respectively and SMOReg has a polynomial kernel. A comprehensive result on 10cv is presented in Table I. Examining Table I, we may therefore draw the following conclusions. First of all, in Table I, we have arranged the feature selection methods based on the number of selected features. The first row of Table I indicate no feature selection (i.e. all 300 features are used), while the other rows are arranged in ascending order of the number of features selected. We shall compare the results of the prediction models arranged in the columns in Table I. The process of feature selection was able to find the most significant features that influenced the drug release rate. It may be observed that feature vectors from all the mentioned feature selection obtained a reduced set of the most influencing features. Therefore, a general theory may be drawn about how and which features most dominating for the PLGA drug release rate.

It is worth mentioning that the best result presented by Szlkeket *et al.* [2] is an RMSE of 15.4 considering 11 selected features using MLP and 17 features with an RMSE of 14.3 using MLP. From Table I, it may be observed that considering all (300 features), the best result we may achieve is by using REP resulting in on RMSE of 13.05 (the average of the 10 CV result). Therefore, any regression model tested on the reduced feature set must compete with this result. In our study, the best result was obtained with the feature set using the wrapperGPgreedy method with RSMES of 14.88, 20.22, 15.20, 13.31, and 20.86 using the GPReg, LReg, MLP, REP and SMOReg elementary models respectively. Therefore, we may consider the features ‘Fused ring count’, ‘Heteroaromatic ring count’, ‘Largest ring system size’, ‘Chain atom count’, ‘Chain bond count’ and ‘Quaternary structure’ from the Protein descriptors group of features,

‘PVA conc inner phase’, ‘PVA conc outer phase’, ‘PVA Mw’, and ‘PLGA to Placticizer’ from the Formulation characteristics group of features, ‘ASA+’, ‘Szedged index’ and ‘pH 12 logd’ from the Plasticizer group of features and from the Emulsifier group, we have ‘a(yy)’ and ‘Time in Days’ as the most influencing feature set.

After obtained the best features, we resorted to using the ensemble techniques mentioned in section II-D. A comprehensive comparison of the results obtained from the ensemble methods and other elementary regression models is given in Figure 2. From the results given in Figure 2, it is evident that few of the ensemble methods significantly improve the results of the elementary regression models in order to compete with the results obtained by the REP for all the features. The average RMSEs obtained using Random subspace using REP, Random subspace using MLP, Random subspace using GPreg, Bagging using REP, Bagging using MLP, MOR, QWOR and EWE are 13.85, 18.20, 18.72, 11.49, 12.30, 10.43, 10.06 and 7.67 respectively.

IV. CONCLUSION

Analysis of the effectiveness of the ensemble methods should be based on comparison of the results obtained by the best elementary predictor. In our case, among the tested simple predictors, the lowest RMSE was reached with REP and was 13.34. Ensemble methods should improve regression accuracy over the best elementary predictor. The EWE ensemble method offers lowest RMSE which proves, that in certain cases, combining the outputs of several predictors allows us to improve the overall accuracy of regression. As discussed in section II-D, it is essential to ensure diversity among the ensemble’s constituents. Among the tested techniques, the best results were obtained by the ensemble of five heterogeneous regression algorithms. Weighting of their outputs is most effective when weights were set using evolutionary-based algorithm described in section II-D. the authors would like to emphasise that they do not claim that it is the best method for creating diversified ensemble regression methods in general, rather it appeared to be the best one for the regression problem under consideration. We suggest that in all the cases, a broad range of experiments with a variety of elementary regression algorithms and ensemble methods should be used in order to find the best solution. Nonetheless, the obtained results proves that the proposed EWE method is one of the effective options to offer solution to the present problem.

ACKNOWLEDGMENT

This work was supported by the IPROCUM Marie Curie initial training network, funded through the People Programme (Marie Curie Actions) of the European Union’s Seventh Framework Programme FP7/2007-2013/ under REA grant agreement No. 316555.

This paper has been elaborated in the framework of the project Opportunity for young researchers, reg. no. CZ.1.07/2.3.00/30.0016, supported by Operational Programme Education for Competitiveness and co-financed by the European Social Fund and the state budget of the Czech Republic.

REFERENCES

- [1] C. E. Astete and C. M. Sabliov, "Synthesis and characterization of plga nanoparticles," *Journal of Biomaterials Science, Polymer Edition*, vol. 17, no. 3, pp. 247–289, 2006.
- [2] J. Szlkek, A. Paclawski, R. Lau, R. Jachowicz, and A. Mendyk, "Heuristic modeling of macromolecule release from plga microspheres," *International journal of nanomedicine*, vol. 8, p. 4601, 2013.
- [3] S. Fredenberg, M. Wahlgren, M. Reslow, and A. Axelsson, "The mechanisms of drug release in poly (lactic-co-glycolic acid)-based drug delivery systemsa review," *International journal of pharmaceutics*, vol. 415, no. 1, pp. 34–52, 2011.
- [4] J. Kang and S. P. Schwendeman, "Pore closing and opening in biodegradable polymers and their effect on the controlled release of proteins," *Molecular pharmaceutics*, vol. 4, no. 1, pp. 104–118, 2007.
- [5] J. Kang, O. Lambert, M. Ausborn, and S. P. Schwendeman, "Stability of proteins encapsulated in injectable and biodegradable poly (lactide-co-glycolide)-glucose millicylinders," *International journal of pharmaceutics*, vol. 357, no. 1, pp. 235–243, 2008.
- [6] M. Blanco and M. Alonso, "Development and characterization of protein-loaded poly (lactide-co-glycolide) nanospheres," *European Journal of Pharmaceutics and Biopharmaceutics*, vol. 43, no. 3, pp. 287–294, 1997.
- [7] R. M. Mainardes and R. C. Evangelista, "Plga nanoparticles containing praziquantel: effect of formulation variables on size distribution," *International journal of pharmaceutics*, vol. 290, no. 1, pp. 137–144, 2005.
- [8] K. Zygourakis and P. A. Markenscoff, "Computer-aided design of bioerodible devices with optimal release characteristics: a cellular automata approach," *Biomaterials*, vol. 17, no. 2, pp. 125–135, 1996.
- [9] A. Gopferich, "Mechanisms of polymer degradation and erosion," *Biomaterials*, vol. 17, no. 2, pp. 103–114, 1996.
- [10] J. Siepmann, N. Faisant, and J.-P. Benoit, "A new mathematical model quantifying drug release from bioerodible microparticles using monte carlo simulations," *Pharmaceutical Research*, vol. 19, no. 12, pp. 1885–1893, 2002.
- [11] M. A. Hall and L. A. Smith, "Practical feature subset selection for machine learning," 1998.
- [12] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1, pp. 273–324, 1997.
- [13] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.
- [14] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (gpml) toolbox," *The Journal of Machine Learning Research*, vol. 9999, pp. 3011–3015, 2010.
- [15] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 1st ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1994.
- [16] P. J. Werbos, "Beyond regression: New tools for prediction and analysis in the behavioral sciences," 1975.
- [17] D. E. Rumelhart and J. L. McClelland, "Parallel distributed processing: explorations in the microstructure of cognition. volume 1. foundations," 1986.
- [18] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The rprop algorithm," in *Neural Networks, 1993., IEEE International Conference on*. IEEE, 1993, pp. 586–591.
- [19] J. R. Quinlan, "Simplifying decision trees," *International journal of man-machine studies*, vol. 27, no. 3, pp. 221–234, 1987.
- [20] W. Mohamed, M. Salleh, and A. Omar, "A comparative study of reduced error pruning method in decision tree algorithms," in *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*, Nov 2012, pp. 392–397.
- [21] A. J. Smola and B. Scholkopf, *Learning with kernels*. Cite-seer, 1998.
- [22] A. J. Smola and B. Schollkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.
- [23] B. Scholkopf, C. J. Burges, and A. J. Smola, *Advances in kernel methods: support vector learning*. MIT press, 1999.
- [24] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in large margin classifiers*, vol. 10, no. 3, pp. 61–74, 1999.
- [25] T. K. Ho, "The random subspace method for constructing decision forests," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 8, pp. 832–844, Aug 1998.
- [26] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [27] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.
- [28] E. Bauer and R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," *Machine learning*, vol. 36, no. 1-2, pp. 105–139, 1999.
- [29] D. E. Goldberg *et al.*, *Genetic algorithms in search, optimization, and machine learning*. Addison-wesley Reading Menlo Park, 1989, vol. 412.